Proceedings of the Thirtieth International Conference on Automated Planning and Scheduling (ICAPS 2020)

Real Time Crowd Navigation from First Principles of Probability Theory

Pete Trautman, Karankumar Patel
Honda Research Institute, San Jose, CA

Abstract

Constructing realistic and real time human-robot interaction models is a core challenge in crowd navigation. In this paper we derive a robot-agent interaction density from first principles of probability theory; we call our approach "first order interacting Gaussian processes" (foIGP). Furthermore, we compute locally optimal solutions—with respect to multifaceted agent "intent" and "flexibility"—in near real time on a laptop CPU. We test on challenging scenarios from the ETH crowd dataset and show that the safety and efficiency statistics of foIGP is competitive with *human* safety and efficiency statistics. Further, we compute the safety and efficiency statistics of dynamic window avoidance, a physics based model variant of foIGP, a Monte Carlo inference based approach, and the best performing deep reinforcement learning algorithm; foIGP outperforms all of them.

1 Introduction

In 2010, (Trautman and Krause 2010) explained the "freezing robot problem" (FRP): for crowd navigation, independently modeling the robot and the humans causes the robot to freeze (or take unnecessary and possibly unsafe evasive maneuvers) as congestion worsens. In (Trautman et al. 2013), the FRP was demonstrated experimentally in a six month cafeteria study by showing that the robot-agent independence assumption causes frequent robot halting and a 3x decrement in safety. Further, the FRP has been reproduced in multiple studies (e.g., (Kretzschmar et al. 2016), (Fan et al. 2019)), was the topic of an academic workshop (Pettre et al. 2018), and has even appeared in the popular media (Adams 2017). Importantly, (Trautman and Krause 2010) provided a remedy for the FRP: "cooperative collision avoidance (CCA)" between the robot and human (first order interaction, or FO) must be modeled. In 2012, the crowd navigation community began adopting FO models (Kuderer et al. 2012); by 2017, FO models for crowd navigation (Chen et al. 2017; 2019; Tai et al. 2018) and prediction (Alahi, Goel, and et al 2016; Vemula, Muelling, and Oh 2018; Gupta et al. 2018; Ivanovich et al. 2018) were commonplace.

Despite the substantial interest, very little is known about what form a cooperative collision avoidance (CCA) function should take. Further, merely *formulating* the joint is insufficient because inference is non-trivial: for a 2D holonomic robot performing polygonal obstacle collision avoidance (decoupled robots and agents), complexity is NP-Hard (Canny and Reif 1987); for continuous time Bayesian networks (similar to crowd navigation), complexity is also NP-hard (Sturlaugson and Sheppard 2014). This complexity is easily visualized: for a planar discretized action space of n_t agents, each agent can move in 8 directions at each time step. For prediction horizon T, then, the system has $8^{n_t T}$ states. Two questions are thus imperative:

- *Which CCA functions best mitigate unnecessary evasive maneuvers (FRP)?*
- *Given a CCA function, how can the* $\arg\max$ *of the robot-agent joint be efficiently computed?*

Most practitioners assume that CCA functions can take any form. However, Theorem 1 shows that for Gaussian process (GP) mixtures the CCA function is constrained: it contains *only* warranted prior information iff it takes as argument *only* the full set of mixture statistics. Theorem 1 thus implies that free parameters are prohibited in CCA functions over GP mixtures (implementation tuning parameters, like prediction horizon and discretization, are permitted). Compared to existing approaches, this is an important result: the social forces model (Helbing and Molnar 1995) contains $4n_t + 1$ free parameters for n_t agents; deep reinforcement learning relies on careful network parameterization and reward function tuning. Minimizing free parameters has practical relevance: invalid CCA functions can lead to skewed performance outside of the evaluation set (e.g., parameter tuning can lead to poor generalization). Section 7 shows that valid CCA functions lead to human-competitive safety and efficiency performance without parameter tuning. These CCA results are our first contribution.

Our second contribution is an efficient ($\approx 0.2s$ replan time) and novel approach to inference of a statistically valid FO joint, called *first order interacting Gaussian processes* or foIGP. We leverage Theorem 1 to determine the valid joint; we then interpret the GP means as optimization variables that are modulated by the GP covariances. Effectively, then, we are optimizing over a *function space*—rather than a state space of time ordered positions—because the dense GP covariance matrix encodes a smooth evolution of trajectories that mutually shape each other. Further, our formulation is compatible with any agent Gaussian mixture model (GMM). For crowd navigation and prediction research, agent models are often GMMs. For example, Gaussian noise dynamical models are GMMs and deep network outputs can be modeled with a GMM (Ivanovich et al. 2018).

Finally, we test the safety and efficiency performance of humans, two traditional planners, the best performing deep reinforcement learning approach, two variants of foIGP, and foIGP on 242 ETH runs. foIGP displays safety-efficiency statistics competitive with humans and significantly outperform the other approaches.

2 Related Work

Roboticists have been investigating navigation in human environments since the 1990s. Two landmark studies were the RHINO (Burgard, Cremers, and et al 1998) and MINERVA (Thrun, Beetz, and et al 2000) experiments, which inspired broad research in navigation near humans. A standard approach to crowd prediction assumes agent independence; (Du Toit and Burdick 2012) observed that this leads to an uncertainty explosion making efficient navigation impossible. A number of researchers thus focused on *controlling* uncertainty. For instance (Thompson and et al 2009) developed high fidelity agent models in the hope that controlled uncertainty would improve navigation. In (Du Toit and Burdick 2012) predictive uncertainty is bounded; intractability is avoided with receding horizon control; collision checking algorithms keep navigation safe. The insight here was that since replanning is used, predictive covariance can be held constant at measurement noise. In (Joseph, Doshi-Velez, and Roy 2011) sophisticated individual agent models are developed: GP mixtures (Rasmussen and Williams 2006) with a Dirichlet Process (DP) prior over mixture weights. DP priors allowed the number of motion patterns to be data driven while the GP enables intermotion variability. However, bounding uncertainty cannot prevent freezing robot behavior (Trautman et al. 2013).

Human intention aware planning is a popular crowd navigation approach (Kruse et al. 2013). Proximity relationship knowledge ("proxemics", (Hall 1966)), provides insight about social robot design. In (Mead, Atrash, and Mataric 2011) proxemics informs navigation protocol. In (Svenstrup, Bak, and Andersen 2010) RRTs are combined with a proxemic potential field. In (Rios-Martinez and et al 2011) personal space rules guide the robot's behavior. In (Unhelkar and et al 2015), anticipatory indicators of human walking informs co-navigation. Although these approaches model human-robot interaction, they do not model human-robot *cooperation*.

We point out that the above algorithms, as well as (Fulgenzi and et al 2007), (van den Berg, Lin, and Manocha 2008) and its relatives such as (Snape et al. 2011) and (Fulgenzi et al. 2008), (Aoude and et al 2011), (Joseph, Doshi-Velez, and Roy 2011) were investigated theoretically and experimentally in Section 5.4 of (Trautman 2013) and found to suffer from the FRP. As the author argues, this list is an exhaustive accounting of pre-2013 crowd navigation algorithms.

Some approaches learn navigation strategies by observing examples. In (Kretzschmar et al. 2016) inverse reinforcement learning finds the robot-crowd joint distribution. In (Ziebart, Maas, and et al 2008), maximum entropy inverse reinforcement learning learns cab driver strategies from data. In (Sadigh et al. 2018) coupled human-robot models generate communicative autonomous behaviors. Using deep reinforcement learning variants for crowd navigation is also highly popular. For example, (Chen et al. 2017; Fan et al. 2019) use a variant of "adversarial reinforcement learning," with the reward encoding cooperative collision avoidance. Typically, the relationship between humans is ignored (e.g., the interaction between robot and human is modeled); importantly, (Chen et al. 2019) models human-human interaction in a method called "socially aware reinforcement learning (SARL)." SARL has the best safety-efficiency profile of the DRL approaches.

While the deep learning approaches in (Pfeiffer et al. 2018; Alahi, Goel, and et al 2016; Gupta et al. 2018; Vemula, Muelling, and Oh 2018) focus on *prediction*, they are an important contribution to crowd navigation. The insight of these papers is that a connected layer corresponds to robot-human interaction; long-short term memory and generative adversarial techniques are used. In (Ivanovich et al. 2018), a variational auto encoder captures multimodality.

3 Terminology

Define $\mathcal{X}$ to be the state space of the crowd and robot. For example, $\mathcal{X}$ could be $\mathbb{R}^2$ for planar navigation. We collect measurements $\mathbf{z}^R_{1:t}$ of the robot trajectory $\mathbf{f}^R\colon t \in \mathbb{R} \to \mathcal{X}$ and n_t measurements $\mathbf{z}^1_{1:t}, \ldots, \mathbf{z}^{n_t}_{1:t}$ of the human trajectories $\mathbf{f} = [\mathbf{f}^1, \ldots, \mathbf{f}^{n_t}]\colon t \in \mathbb{R} \to \mathcal{X}$, where $\mathbf{f}$ represents the crowd. The functions $\mathbf{f}^R$ and $\mathbf{f}$ are governed by $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t})$ and $p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$ for each i. We use the shorthand $\mathbf{z}^{\mathbf{f}}_{1:t} = [\mathbf{z}^{\mathbf{f}^1}_{1:t}, \ldots, \mathbf{z}^{\mathbf{f}^{n_t}}_{1:t}]$. We do not assume that $\mathbf{z}^R_{1:t}$ or $\mathbf{z}^{\mathbf{f}}_{1:t}$ are complete, i.e. for any $\tau \in 1:t$ or $i \in 1:n_t$ some $\mathbf{z}^R_\tau$ or $\mathbf{z}^{\mathbf{f}^i}_\tau$ could be missing. We say that $\mathbf{f}^R, \mathbf{f}^i \in \mathcal{F}(\mathcal{X})$, the function space over $\mathcal{X}$. We use the shorthand $\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) \equiv GP(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$ to define our agent GP mixture models:

$$\begin{aligned} p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t}) &= \sum_{k_i=1}^{N_t^{\mathbf{f}^i}} w^{\mathbf{f}^i}_{k_i} \mathcal{N}(\mathbf{f}^i \mid \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}, \boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i}) \quad \text{(crowd agent)} \\ p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}) &= \sum_{\ell=1}^{N_t^R} w^R_\ell \mathcal{N}(\mathbf{f}^R \mid \boldsymbol{\mu}^R_\ell, \boldsymbol{\Sigma}^R_\ell). \qquad \text{(robot)} \end{aligned} \tag{3.1}$$

The mixture weights, means and covariances update at each time step, but we suppress time for clarity: $w^{\mathbf{f}^i}_{k_i} \equiv w^{\mathbf{f}^i}_{k_i}(t)$, $\boldsymbol{\mu}^{\mathbf{f}^i}_{k_i} \equiv \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}(t)$, and $\boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i} \equiv \boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i}(t)$.

Definition 1 (GP shorthand). *Let* $\mathcal{N}_{\mathbf{f}^R,\ell} = \mathcal{N}(\mathbf{f}^R \mid \boldsymbol{\mu}^R_\ell, \boldsymbol{\Sigma}^R_\ell)$. *For each* $\mathbf{f}^i$, *let* $\mathcal{N}_{\mathbf{f}^p,v} = \mathcal{N}(\mathbf{f}^p \mid \boldsymbol{\mu}^p_v, \boldsymbol{\Sigma}^p_v)$ *where* $v = k_i$ *and* $p \in \{1, \ldots, n_t\}$.

For example $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}) = \sum_{\ell=1}^{N_t^R} w^R_\ell \mathcal{N}_{\mathbf{f}^R,\ell}$.

Definition 2. *We call* $\boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}, \boldsymbol{\mu}^R_\ell$ *the crowd agent and robot* **intents**. *If* $N_t^{\mathbf{f}^i}$ *or* N_t^R *is greater than 1,* **intent ambiguity** *is present.* **Intent preferences** *are* $w^{\mathbf{f}^i}_{k_i}$ *and* w^R_ℓ.

Consider Figure 1 where $p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t}) = \sum_{k_i=L,R} w^{\mathbf{f}^i}_{k_i} \mathcal{N}_{\mathbf{f}^i,k_i}$. Because $N_t^{\mathbf{f}^i} > 1$, intent ambi-

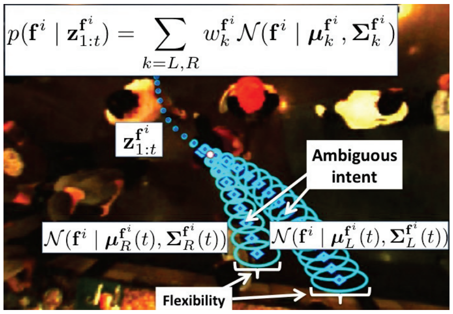

$p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t}) = \sum_{k=L,R} w^{\mathbf{f}^i}_k \mathcal{N}(\mathbf{f}^i \mid \boldsymbol{\mu}^{\mathbf{f}^i}_k, \boldsymbol{\Sigma}^{\mathbf{f}^i}_k)$
$\mathbf{z}^{\mathbf{f}^i}_{1:t}$
Ambiguous intent
$\mathcal{N}(\mathbf{f}^i \mid \boldsymbol{\mu}^{\mathbf{f}^i}_R(t), \boldsymbol{\Sigma}^{\mathbf{f}^i}_R(t))$
$\mathcal{N}(\mathbf{f}^i \mid \boldsymbol{\mu}^{\mathbf{f}^i}_L(t), \boldsymbol{\Sigma}^{\mathbf{f}^i}_L(t))$
Flexibility

Figure 1: GP components model intent to go $\boldsymbol{\mu}^{\mathbf{f}^i}_R, \boldsymbol{\mu}^{\mathbf{f}^i}_L$ (with preferences $w^{\mathbf{f}^i}_R$ and $w^{\mathbf{f}^i}_L$). Within a component, agent has flexibility $\boldsymbol{\Sigma}^{\mathbf{f}^i}_{R,L}$ about $\boldsymbol{\mu}^{\mathbf{f}^i}_{R,L}$.

guity is present. Intent preferences are the left and right values $w^{\mathbf{f}^i}_L$ and $w^{\mathbf{f}^i}_R$.

Definition 3. Flexibility *is agent willingness to compromise about intent. Mathematically, the flexibility of intent $\boldsymbol{\mu}$ is $\boldsymbol{\Sigma}$.*

Flexibility is motivated by the following: imagine that we draw a sample $\mathbf{x}$ and evaluate $\mathcal{N}(\mathbf{f}^i = \mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$. For large covariance, even if $\mathbf{x}$ is far from the intent $\boldsymbol{\mu}$ the probability of $\mathbf{x}$ is nonzero (e.g., the mixture component has "large flexibility"). Using these variables, we define the following.

Problem Formulation Generative crowd navigation uses a joint robot-crowd density $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}^R_{1:t}, \mathbf{z}^{\mathbf{f}}_{1:t})$ to generate the action $u_t = \mathbf{f}^{R*}_{t+1}$ at time t according to

$$[\mathbf{f}^R, \ldots, \mathbf{f}^{n_t}]^* = \underset{\mathbf{f}^R, \ldots, \mathbf{f}^{n_t}}{\arg\max}\, p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}^R_{1:t}, \mathbf{z}^{\mathbf{f}}_{1:t}). \tag{3.2}$$

Starting with GP mixture agent models, we derive a set of principles that any interaction function—the function coupling the agent models—must obey. We then propose a valid interaction function that has desirable safety and efficiency properties and an optimization routine to find $u_t = \mathbf{f}^{R*}_{t+1}$.

4 Statistical Principles of First Order Interacting GPs

In (Trautman 2017) the following factorization was studied:

$$p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}^R_{1:t}, \mathbf{z}^{\mathbf{f}}_{1:t}) = \psi(\mathbf{f}^R, \mathbf{f}, \gamma) p(\mathbf{f} \mid \mathbf{z}^{\mathbf{f}}_{1:t}) p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t})$$
$$= \prod_{i=1}^{n_t} \psi(\mathbf{f}^R, \mathbf{f}^i, \gamma) p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t}) p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}). \tag{4.1}$$

The function $\psi(\mathbf{f}^R, \mathbf{f}, \gamma)$, $\gamma \in \mathbb{R}$ is a product of pairwise interaction functions $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma)$ modulated by $p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$. This model captures robot-agent interaction but ignores inter-human interaction (while including human-human interaction is straightforward in definition 4.1, implementation is substantially different and more complex; we leave that work for a future paper). We study factorization 4.1.

We start with the case of two agent interaction ($n_t = 1$, 1 robot) We intend to determine statistically valid forms of $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma)$. Since $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}), p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$ encode *online* intention and flexibility information (Definitions 2, 3), the means and covariances capture inter-agent intention and flexibility that is *specific to and influenced by* agent interaction. If the interaction function has finite support (e.g., $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma) = \prod_{t=1}^{T} \left[1 - \exp\left(-\frac{1}{2\gamma}(\mathbf{f}^R_t - \mathbf{f}^i_t)\right)\right]$ where $\gamma > 0$), joint flexibility is altered *in a static and unwarranted way* by γ (Figure 2); any probability mass in $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma)$ alters the agent-specific flexibilities in $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t})$ and $p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$.

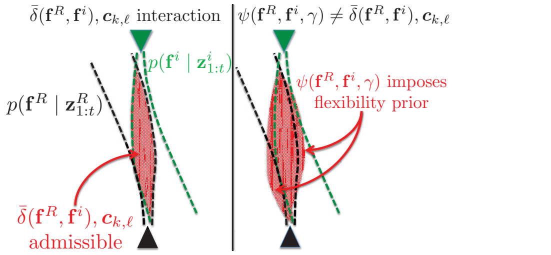

$\bar{\delta}(\mathbf{f}^R, \mathbf{f}^i), \boldsymbol{c}_{k,\ell}$ interaction
$\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma) \neq \bar{\delta}(\mathbf{f}^R, \mathbf{f}^i), \boldsymbol{c}_{k,\ell}$
$p(\mathbf{f}^i \mid \mathbf{z}^i_{1:t})$
$p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t})$
$\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma)$ imposes flexibility prior
$\bar{\delta}(\mathbf{f}^R, \mathbf{f}^i), \boldsymbol{c}_{k,\ell}$
admissible

Figure 2: Human in green, robot in black, dotted lines one standard deviation; $\bar{\delta}(\mathbf{f}^R, \mathbf{f}), \boldsymbol{c}_{k,\ell}$ does not alter agent statistics; $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma) \neq \bar{\delta}(\mathbf{f}^R, \mathbf{f}), \boldsymbol{c}_{k,\ell}$ *imposes flexibility prior* on $p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t}), p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t})$.

In other words, since $\mathbf{f}^R, \mathbf{f}^i$ are both governed by GP covariances, we already have a measure of how much each variable can deform (e.g., our model of agent compromise is the covariance). If we include some factor γ—perhaps tuned to a specific dataset—we impose unwarranted flexibility on the $\mathbf{f}^R, \mathbf{f}^i$ system. While it may be the case that agent flexibility changes during interaction, it is unclear how to measure this. We emphasize that a promising area of study is to model how individual flexibility changes during interaction.

To preserve the statistics of $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}), p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$ we introduce the transform $\bar{\delta}(\mathbf{f}^R, \mathbf{f})$, where $\bar{\delta}(\mathbf{f}^R, \mathbf{f}) = 1$ if samples of $\mathbf{f}^R, \mathbf{f}^i$ are not equal at any time t and $\bar{\delta}(\mathbf{f}^R, \mathbf{f}) = 0$ otherwise. We also introduce $\boldsymbol{c}_{k_i,\ell} = \boldsymbol{c}(\boldsymbol{\mu}^R_\ell, \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}, \boldsymbol{\Sigma}^R_\ell, \boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i})$, a function of the GP means and covariances. Using $\bar{\delta}(\mathbf{f}^R, \mathbf{f})$ and $\boldsymbol{c}_{k_i,\ell}$ we improve the proof from (Trautman 2017):

Theorem 1. *If $p(\mathbf{f}^R \mid \mathbf{z}^R_{1:t}), p(\mathbf{f}^i \mid \mathbf{z}^{\mathbf{f}^i}_{1:t})$ are GP mixtures then $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma) = \bar{\delta}(\mathbf{f}^R, \mathbf{f}^i)$ or a set of constants $\boldsymbol{c}_{k_i,\ell} = \boldsymbol{c}(\boldsymbol{\mu}^R_\ell, \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}, \boldsymbol{\Sigma}^R_\ell, \boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i})$. That is, if $\psi(\mathbf{f}^R, \mathbf{f}^i, \gamma)$ has non-zero support over $\mathbf{f}^R$ or $\mathbf{f}^i$ then it acts as an unwarranted prior.*

Proof. See the supplement for the proof. □

This theorem tells us that we have two choices for the coefficients of each GP product in the joint mixture. If we used a trajectory basis so that $p(\mathbf{f}^R, \mathbf{f}^i \mid \mathbf{z}_{1:t}) = \sum_{g=1}^{G} w_g \delta([\mathbf{f}^R, \mathbf{f}^i] - [\mathbf{f}^R, \mathbf{f}^i]_g)$ then $\bar{\delta}(\mathbf{f}^R, \mathbf{f})$ would be applicable. However, since the interaction of two GPs *is probabilistic*, a "coupling" probability is appropriate.

Corollary 2. *The coefficient $\boldsymbol{c}_{k_i,\ell}$ does not impose unwarranted prior information iff it takes as argument* only *the full set of statistics of the agent models.*

Proof. See supplement for proof. □

In summary, if parameters are added to or removed from $\boldsymbol{c}(\boldsymbol{\mu}_\ell^R, \boldsymbol{\mu}_{k_i}^{\mathbf{f}^i}, \boldsymbol{\Sigma}_\ell^R, \boldsymbol{\Sigma}_{k_i}^{\mathbf{f}^i})$ then the model is statistically invalid. For instance, modifying the covariance changes the model's predictive confidence (or, equivalently, agent flexibility) in some unwarranted way.

5 Derivation of First Order Interacting GPs

For GPs, the probability of robot-agent collision does not only involve time-aligned terms (Du Toit and Burdick 2012)

$$Z_{\ell,k_i,t}^{-1} = \int \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\mu}_{\ell,t}^R, \boldsymbol{\sigma}_{R,\ell}^{t,t}\right) \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\mu}_{k_i,t}^{\mathbf{f}^i}, \boldsymbol{\sigma}_{\mathbf{f}^i,k_i}^{t,t}\right)$$

where $\boldsymbol{\mu}_{\ell,t}^R = \boldsymbol{\mu}_\ell^R(t), \boldsymbol{\mu}_{k_i,t}^{\mathbf{f}^i} = \boldsymbol{\mu}_{k_i}^{\mathbf{f}^i}(t) \in \mathbb{R}^2$ and $\boldsymbol{\sigma}^{t,t}$ is the t'th diagonal of $\boldsymbol{\Sigma}$. Since $\boldsymbol{\Sigma}_\ell^R, \boldsymbol{\Sigma}_{k_i}^{\mathbf{f}^i}$ are dense, positions are correlated via covariance off diagonals and

$$\begin{aligned}&\int \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\mu}_{\ell,t}^R, \boldsymbol{\sigma}_{R,\ell}^{t,\tau}\right) \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\mu}_{k_i,\tau}^{\mathbf{f}^i}, \boldsymbol{\sigma}_{\mathbf{f}^i,k_i}^{t,\tau}\right)\\ &= w_{\ell,k_i,t} \exp\left[-\frac{1}{2}(\boldsymbol{\mu}_{\ell,t}^R - \boldsymbol{\mu}_{k_i,\tau}^{\mathbf{f}^i})^\top (\boldsymbol{\sigma}_{\ell+k_i}^{t,\tau}\mathbb{I})^{-1}(\boldsymbol{\mu}_{\ell,t}^R - \boldsymbol{\mu}_{k_i,\tau}^{\mathbf{f}^i})\right]\\ &\equiv w_{\ell,k_i,t} Z_{\ell,k_i,t,\tau}^{-1}, x\end{aligned}$$

contributes to collision probability ($\boldsymbol{\sigma}_{\ell+k_i}^{t,\tau}$ is the (t,τ)'th element of $\boldsymbol{\Sigma}_{\ell+k_i} = \boldsymbol{\Sigma}_\ell^R + \boldsymbol{\Sigma}_{k_i}^{\mathbf{f}^i}$, $w_{\ell,k_i,t} = (2\pi\boldsymbol{\sigma}_{\ell+k_i}^{t,\tau})^{-1/2}$). Since $Z_{\ell,k_i,t,\tau}^{-1}$ is the coupling between $\mathbf{f}_t^R$ and $\mathbf{f}_\tau^i$, the value $\prod_{\tau=1}^T(1 - Z_{\ell,k_i,t,\tau}^{-1})$ is the *decoupling* between the robot at t and the *trajectory* of agent i.

Definition 4. *The symbol $\mathbb{P}(\neg\kappa)$—the probability of not colliding—represents the* **decoupling** *of $\mathcal{N}_{\mathbf{f}^R,\ell}$ and $\mathcal{N}_{\mathbf{f}^i,k_i}$*

$$\mathbb{P}(\neg\kappa) = \prod_{t=1}^T \prod_{\tau=1}^T (1 - Z_{\ell,k_i,t,\tau}^{-1}). \tag{5.1}$$

Definition 5. *The transform $\mathbf{P}_{\neg\kappa}$ measures how* **decoupled** *the robot and agent GPs $\mathcal{N}_{\mathbf{f}^R,\ell}$ and $\mathcal{N}_{\mathbf{f}^i,k_i}$ are:*

$$\begin{aligned}\mathbf{P}_{\neg\kappa} : \mathcal{N}_{\mathbf{f}^R,\ell}\mathcal{N}_{\mathbf{f}^i,k_i} &\to \prod_{t=1}^T \prod_{\tau=1}^T (1 - Z_{\ell,k_i,t,\tau}^{-1})\mathcal{N}_{\mathbf{f}^R,\ell}\mathcal{N}_{\mathbf{f}^i,k_i}\\ &= \boldsymbol{\Lambda}_{\ell,k_i}\mathcal{N}_{\mathbf{f}^R,\ell}\mathcal{N}_{\mathbf{f}^i,k_i}.\end{aligned}$$

where $\boldsymbol{\Lambda}_{\ell,k_i} \equiv \prod_{t=1}^T \prod_{\tau=1}^T (1 - Z_{\ell,k_i,t,\tau}^{-1})$.

The quantity $\boldsymbol{\Lambda}_{\ell,k_i}$ is close to 1 when $\mathcal{N}_{\mathbf{f}^R,\ell}$ is decoupled from $\mathcal{N}_{\mathbf{f}^i,k_i}$ and near zero when $\mathcal{N}_{\mathbf{f}^R,\ell}$ is coupled to $\mathcal{N}_{\mathbf{f}^i,k_i}$ (Figure 3); since $\boldsymbol{\Lambda}_{\ell,k_i}$ changes based on $\boldsymbol{\mu}_\ell^R$ and $\boldsymbol{\mu}_{k_i}^{\mathbf{f}^i}$ it is a natural cooperative collision avoidance metric.

Corollary 3. *The transform $\mathbf{P}_{\neg\kappa}$ does not impose unwarranted prior information on the GP mixture components.*

We now define single agent first order IGP.

Definition 6 (**Single agent foIGP**)**.** *Combining Equations 4.1, 3.1, and Corollary 3,* **single agent foIGP** *is*

$$\begin{aligned}p(\mathbf{f}^R, \mathbf{f}^i \mid \mathbf{z}_{1:t}) &= \mathbf{P}_{\neg\kappa}\left[\sum_{\ell=1}^{N_t^R} w_\ell^R \mathcal{N}_{\mathbf{f}^R,\ell} \sum_{k_i=1}^{N_t^{\mathbf{f}^i}} w_{k_i}^{\mathbf{f}^i}\mathcal{N}_{\mathbf{f}^i,k_i}\right]\\ &= \sum_{\ell=1}^{N_t^R}\sum_{k_i=1}^{N_t^{\mathbf{f}^i}} \boldsymbol{\Lambda}_{\ell,k_i} w_\ell^R w_{k_i}^{\mathbf{f}^i}\mathcal{N}_{\mathbf{f}^R,\ell}\mathcal{N}_{\mathbf{f}^i,k_i}.\end{aligned} \tag{5.2}$$

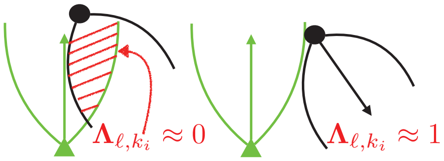

$\boldsymbol{\Lambda}_{\ell,k_i} \approx 0$
$\boldsymbol{\Lambda}_{\ell,k_i} \approx 1$

Figure 3: L: If $\mathcal{N}_{\mathbf{f}^R,\ell}$ and $\mathcal{N}_{\mathbf{f}^i,k_i}$ are tightly coupled, the coefficient $\boldsymbol{\Lambda}_{\ell,k_i} \approx 0$. **R:** When agents are decoupled $\boldsymbol{\Lambda}_{\ell,k_i} \approx 1$.

Leveraging Definition 6 we generalize foIGP to $n_t \geq 1$.

Definition 7. Multi-agent foIGP *is*

$$\begin{aligned}p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}) &= \mathbf{P}_{\neg\kappa}^{IGP}\left[\sum_{\ell=1}^{N_t^R} w_\ell^R\mathcal{N}_{\mathbf{f}^R,\ell}\prod_{i=1}^{n_t}\sum_{k_i=1}^{N_t^{\mathbf{f}^i}} w_{k_i}^{\mathbf{f}^i}\mathcal{N}_{\mathbf{f}^i,k_i}\right]\\ &= \mathbf{P}_{\neg\kappa}^{IGP}\Bigg[\sum_{\ell=1}^{N_t^R} w_\ell^R\mathcal{N}_{\mathbf{f}^R,\ell}\\ &\qquad \times \sum_{k_1=1}^{N_t^{\mathbf{f}^1}} w_{k_1}^{\mathbf{f}^1}\mathcal{N}_{\mathbf{f}^1,k_1} \times \cdots \times \sum_{k_{n_t}=1}^{N_t^{\mathbf{f}^{n_t}}} w_{k_{n_t}}^{\mathbf{f}^{n_t}}\mathcal{N}_{\mathbf{f}^{n_t},k_{n_t}}\Bigg]\\ &= \sum_{\eta=1}^{N_{BIG}} [\boldsymbol{w\Lambda}]_\eta [\mathcal{N}_{\mathbf{f}^R}\mathcal{N}_{\mathbf{f}^1}\cdots\mathcal{N}_{\mathbf{f}^{n_t}}]_\eta\end{aligned} \tag{5.3}$$

where $N_{BIG} = N_t^R\prod_{i=1}^{n_t} N_t^{\mathbf{f}^i}$ and η enumerates all products of robot and agent GPs. The coefficients

$$[\boldsymbol{w\Lambda}]_\eta = \boldsymbol{\Lambda}_\eta^{R,\mathbf{f}^1}\cdots\boldsymbol{\Lambda}_\eta^{R,\mathbf{f}^{n_t}} w_\eta^R w_\eta^{\mathbf{f}^1}\cdots w_\eta^{\mathbf{f}^{n_t}}$$

weight each GP basis element

$$[\mathcal{N}_{\mathbf{f}^R}\mathcal{N}_{\mathbf{f}^1}\cdots\mathcal{N}_{\mathbf{f}^{n_t}}]_\eta = \mathcal{N}_{\mathbf{f}^R,\eta}\mathcal{N}_{\mathbf{f}^1,\eta}\cdots\mathcal{N}_{\mathbf{f}^{n_t},\eta}$$

according to $\mathbf{P}_{\neg\kappa}^{IGP}$. The operator $\mathbf{P}_{\neg\kappa}^{IGP}$ operates pairwise

$$\mathbf{P}_{\neg\kappa}^{IGP} \equiv (\mathbf{P}_{\neg\kappa})^{n_t} = \mathbf{P}_{\neg\kappa}\circ\mathbf{P}_{\neg\kappa}\circ\cdots\circ\mathbf{P}_{\neg\kappa},$$

since we only measure robot-agent interaction.

6 Inference of First Order Interacting GPs

While Equation 5.3 provides a form for $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$, it does not provide content. In particular, if we construct agent distributions (Equations 3.1) in *isolation* of each other, we are unlikely to find large $[\boldsymbol{w\Lambda}]_\eta$. For example, MC_1e5 (see Tables 1 and 2) draws 10^5 joint samples and then takes the sample with largest $[\boldsymbol{w\Lambda}]_\eta$ as the robot action. Despite the computation, MC_1e5 produces unsafe and long trajectories. Instead of brute force enumeration, we seek the $N^* \ll N_{BIG}$ modes that best capture Equation 5.3.

We begin by treating $\boldsymbol{\mu}_\ell^R, \boldsymbol{\mu}_{k_i}^{\mathbf{f}^i}$ as functions $\mathbf{x}_{R,\ell}, \mathbf{x}_{\mathbf{f}^i,k_i} \in \mathcal{F}(\mathbb{R}) \to \mathbb{R}^2$ mapping time to (x,y) position and search for the $\mathbf{x}^*_{R,\ell}, \mathbf{x}^*_{\mathbf{f}^i,k_i}$ that optimize $[\boldsymbol{w}\boldsymbol{\Lambda}]_\eta$. We define

$$
\begin{aligned}
&w_{\mathbf{x}_{R,\ell}} = \mathcal{N}\left(\mathbf{x}_{R,\ell} \mid \boldsymbol{\mu}^R_\ell, \boldsymbol{\Sigma}^R_\ell\right)\\
&w_{\mathbf{x}_{\mathbf{f}^i,k_i}} = \mathcal{N}\left(\mathbf{x}_{\mathbf{f}^i,k_i} \mid \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}, \boldsymbol{\Sigma}^{\mathbf{f}^i}_{k_i}\right)\\
&Z^{-1}_{\mathbf{x}_{R,\ell,t},\mathbf{x}_{\mathbf{f}^i,k_i,\tau}} =\\
&\exp\left[-\frac{1}{2}(\mathbf{x}_{R,\ell,t} - \mathbf{x}_{\mathbf{f}^i,k_i,\tau})^\top(\boldsymbol{\sigma}^{t,\tau}_{\ell+k_i}\mathbb{I})^{-1}(\mathbf{x}_{R,\ell,t} - \mathbf{x}_{\mathbf{f}^i,k_i,\tau})\right].
\end{aligned}
$$

where $\mathbf{x}_{R,\ell,t} = \mathbf{x}_{R,\ell}(t), \mathbf{x}_{\mathbf{f}^i,k_i,\tau} = \mathbf{x}_{\mathbf{f}^i,k_i}(\tau) \in \mathbb{R}^2$.

Definition 8. *Let* $\mathbf{x}_\mathbf{f} = [\mathbf{x}_{\mathbf{f}^1,k_1}, \ldots, \mathbf{x}_{\mathbf{f}^{n_t},k_{n_t}}]$. *Then*

$$
\begin{aligned}
&\boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f}) \equiv\\
&\qquad w_{\mathbf{x}_{R,\ell}} \prod_{i=1}^{n_t}\prod_{t=1}^{T}\prod_{\tau=1}^{T}\left(1 - Z^{-1}_{\mathbf{x}_{R,\ell,t},\mathbf{x}_{\mathbf{f}^i,k_i,\tau}}\right) w_{\mathbf{x}_{\mathbf{f}^i,k_i}}.
\end{aligned}
$$

We use the logarithm to improve numerical accuracy.

Definition 9. *Let*

$$
\begin{aligned}
\log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f}) = &\sum_{i=1}^{n_t}\sum_{t=1}^{T}\sum_{\tau=1}^{T} \log\left(1 - Z^{-1}_{\mathbf{x}_{R,\ell,t},\mathbf{x}_{\mathbf{f}^i,k_i,\tau}}\right)\\
&- \frac{1}{2}(\mathbf{x}_{R,\ell} - \boldsymbol{\mu}^R_\ell)^\top(\boldsymbol{\Sigma}^R)^{-1}(\mathbf{x}_{R,\ell} - \boldsymbol{\mu}^R_\ell)\\
&- \sum_{i=1}^{n_t}\frac{1}{2}(\mathbf{x}_{\mathbf{f}^i,k_i} - \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i})^\top(\boldsymbol{\Sigma}^{\mathbf{f}^i})^{-1}(\mathbf{x}_{\mathbf{f}^i,k_i} - \boldsymbol{\mu}^{\mathbf{f}^i}_{k_i}).
\end{aligned}
\tag{6.1}
$$

We seek $\arg\max_{\mathbf{x}_{R,\ell},\mathbf{x}_\mathbf{f}} \log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f})$. To understand the behavior of $\log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f})$, note that the triple product rewards cooperative collision avoidance while the quadratics penalize solutions that deviate from agent intent.

Unfortunately, $\log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f})$ is non-convex. However, by using Newton's method and carefully chosen initializations we recover strong performance (see Section 7). We first tried automatic differentiation but it was exceedingly slow. Instead, we used hand coded derivatives (see the supplement) and autograd for numerical validation. Finally, since trajectories near the mean ± standard deviation multiples have high probability, we seed the optimizations with $(\boldsymbol{\mu}^R, \boldsymbol{\mu}^\mathbf{f}) \pm (0, [\sigma^R, \sigma^\mathbf{f}], 2[\sigma^R, \sigma^\mathbf{f}], 3[\sigma^R, \sigma^\mathbf{f}])$, where $\sigma^R = \sqrt{\mathrm{diag}(\boldsymbol{\Sigma}^R)}$, $\sigma^\mathbf{f} = \sqrt{\mathrm{diag}(\boldsymbol{\Sigma}^\mathbf{f})}$. Numerical experiments support this idea (see Section 7).

Additionally, Newton optimization provides insight about agents most important to the optimization. In particular, we computed the first step of the Newton optimization for each agent and the robot and then computed the *effective sample size* (ESS) (Doucet and Johansen 2008) of n_t robot-agent pairs to determine how many agents are statistically significant to the optimization. Importantly, ESS finds the significant agents at each time step. A shortcoming is that by only considering robot-agent pairs, we risk ignoring inter-agent effects. Letting H be the Hessian and ∇ the gradient, let

$$
\begin{aligned}
&\Delta\mathbf{x}_{\ell,k_i} =\\
&\quad\left\|\left[H(\log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_{\mathbf{f}^i,k_i}))\right]^{-1}\nabla(\log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_{\mathbf{f}^i,k_i}))\right\|
\end{aligned}
$$

be the norm of the first Newton step.

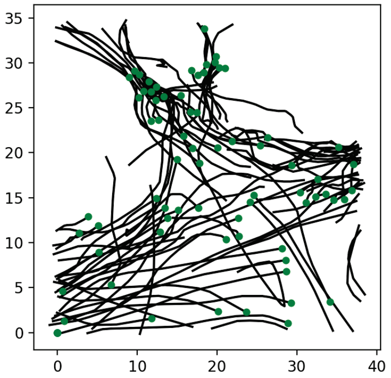


Figure 4: First frame of the ETH data evaluated. Pedestrian current position in green; next 40 time steps plotted as black curves. While other crowd datasets are useful, this subset of ETH has the highest pedestrian density and frequency of interaction; every agent thus provides a challenging navigation negotiation test.

Definition 10 (ESS). *Let* $\overline{\Delta\mathbf{x}_{\ell,k_i}}$ *be the normalization of* $\Delta\mathbf{x}_{\ell,k_i}$. *Then compute*

$$
ESS = \frac{1}{\sum_{j=1}^{n_t}\sum_{\ell,k_j=1}^{N^R_t,N^{\mathbf{f}^i}_t}\overline{\Delta\mathbf{x}_{\ell,k_i}}^2} \leq N^R_t\prod_{i=1}^{n_t}N^{\mathbf{f}^i}_t.
$$

Finally, we define our action protocol (c.f. Equation 3.2):

Definition 11 (Generative crowd planning). *At time* t *compute* $\arg\max_{\mathbf{x}_{R,\ell},\mathbf{x}_\mathbf{f}} \log \boldsymbol{\lambda}_{n_t}(\mathbf{x}_{R,\ell}, \mathbf{x}_\mathbf{f})$ *using the top ESS agents and issue the command* $u^R_t = \mathbf{x}^*_{R,\ell}(t+1)$.

7 Evaluation

Rationale for dataset We considered the crowd datasets ETH (Pellegrini et al. 2009) and UCY (Lerner, Chrysanthou, and Lischinski 2007) and the crowd simulators PEDSIM (Gloor 2016) and Menge (Curtis, Best, and Manocha 2016). Our concern with simulation is that overly aggressive robot behavior is often permitted. For example, our Monte Carlo IGP produced zero collisions in PEDSIM, whereas in this study it was unsafe in 17% of runs (row 6, Table 1). The study in (Chen et al. 2019) observed zero "socially aware reinforcement learning (SARL)" collisions, whereas we observed unsafe runs 17% of the time (row 7, Table 1). These elevated performance statistics are due to the overly flexible simulated humans. From an evaluation perspective, this makes reporting vulnerable to false positives. The agents can be more aggressively tuned; however, we then risk *tuning the simulation to optimize algorithm performance*.

Alternatively, many trajectories in the ETH and UCY datasets have little or no interaction. We can tell that ETH and UCY is mostly linear by looking at the results in recent studies that benchmarked over the whole of both datasets (Alahi, Goel, and et al 2016), (Gupta et al. 2018): linear extrapolation error rates were very low (e.g., 0.39m "average displacement error" over a 5 second prediction horizon for the hotel dataset). This indicates that the dataset

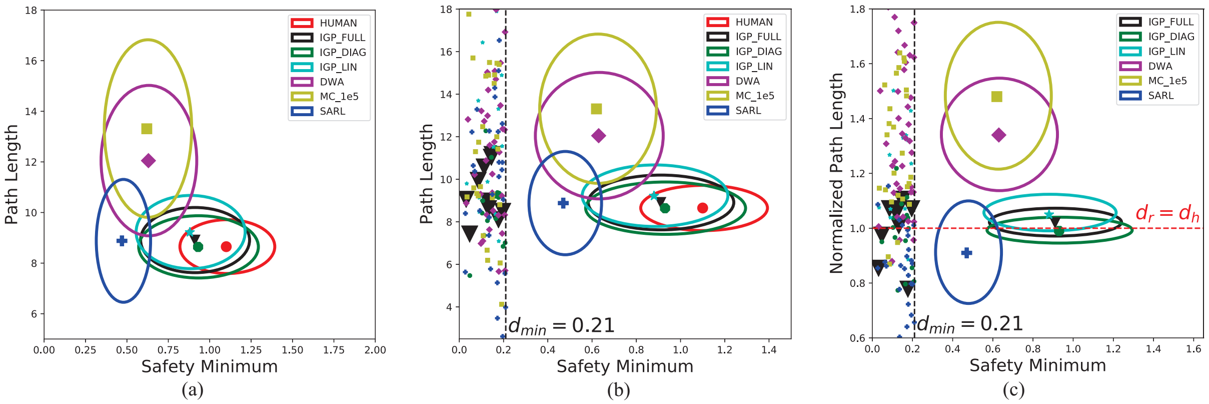


Figure 5: Partial trajectory statistics. Total number of runs is 181; x, y axes in meters. All x-axes plot distance to nearest pedestrian; (a) plots means ± 1 standard deviation of algorithm and human; (b) appends plot (a) with d_{min} threshold, the closest distance two humans passed in the dataset. Inspection of region left of d_{min} shows numerous instances of DWA, MC_1e5, and SARL; (c) normalizes algorithm path length with human path length. E.g., values below $d_r = d_h$ mean the robot moved to goal more directly than human.

	$\mu(s)$	$\%_{s<0.3}$	$\%_{s<0.21}$	$\mu(d_r)$	$\%_{d_r/d_h>1.25}$	$\max(d_r/d_h)$	$\mu(t)$	$\mu(\rho)$
Human	$1.1 \pm .2m$	1.6%	0%	$8.7 \pm 1.0m$	NA	NA	NA	.22±.15
IGP_Full	$.9 \pm .3m$	9%	5%	$8.9 \pm 1.2m$	1%	1.3	$.2 \pm .15s$	.23± .12
IGP_Diag	$.9 \pm .4m$	18%	9%	$8.6 \pm 1.3m$	2%	1.2	$.13 \pm .11s$	.19± .3
IGP_Lin	$.9 \pm .3m$	13%	8%	$9.2 \pm 1.5m$	8%	1.5	$.14 \pm .07s$	.2 ± .14
DWA	$.6 \pm .4m$	35%	24%	$12.1 \pm 2.5m$	48%	4.1	$.1 \pm .03s$	.23 ± .1
MC_1e5	$.6 \pm .3m$	30%	17%	$13.3 \pm 3.5m$	62%	3.9	$2.3 \pm 2.5s$	.2 ± .3
SARL*	$.4 \pm .15m$	37%	17%	$8.3 \pm 2.2m$	1%	1.5	$1.3 \pm .13s$	.22 ± .1

*SARL was trained in five different environments; we report the best performing network.

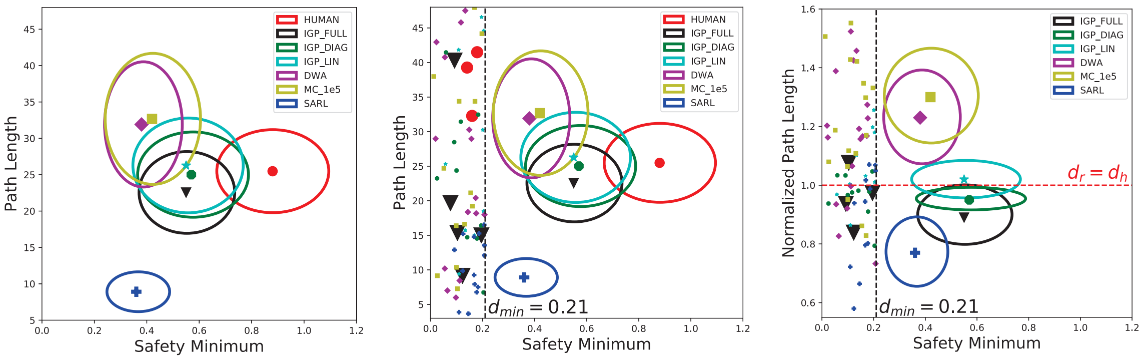
Table 1: Partial trajectory metrics. For each run, distance to nearest pedestrian is s and $\mu(s)$ is the mean; $\%_{s<0.3}, \%_{s<0.21}$ are the percent of runs that $s < 0.3m, 0.21m$; $\mu(d_r)$ is mean robot path length d_r over all runs; $\%_{d_r/d_h>1.25}$ is percent of runs that d_r was 1.25 times human path length d_h; $\mu(t)$ is mean time of all replanning steps; and $\mu(\rho)$ is mean density over all runs (density is number of people in a 3m radius circle around robot, in $people/m^2$).

itself is, in total, too simple. Thus, testing against all of ETH runs the risk of being non-discriminative because straight line solutions are often available (i.e., all algorithms would have high efficiency and low collision rates).

However, a subsample of the ETH dataset (Figure 4; 100 frames, 150 pedestrians) collected for training a deep network in (Ivanovich and Pavone 2019) has many interactions and substantial congestion; indeed, every pedestrian interacts at least once and most pedestrians interact many times during the 100 frame sequence. Ultimately, we chose this subset of ETH so that evaluation would be discriminative. By partitioning our dataset into partial and full trajectories we generated 214 test runs, which provides enough statistical power to draw conclusions. Conversely, if we had added numerous weak interaction runs, statistical power would *decrease* (all algorithms would find the straight line solution).

Further, canned crowd datasets do not immediately suggest a *navigation* testing protocol (ETH and UCY are typically used to benchmark *prediction* algorithms, where testing protocol is straightforward). To derive a navigation test protocol, we expand on an idea from the experimental section of (Trautman and Krause 2010): 1) identify a pedestrian, 2) extract the start and end position of that pedestrian, 3) remove that pedestrian from the observation dataset of the navigation algorithm, and 4) provide the start and end positions of the removed pedestrian and the current and previous positions of the remaining agents to the navigation algorithm. Thus we assure that at least one path through the crowd exists (the one taken by the removed pedestrian). Additionally, by providing the navigation algorithm with start and end points that are joined by a path *through* the crowd, the navigation algorithm naturally confronts high crowd densities ($\mu(\rho)$ column, Tables 1 and 2). Finally, this testing protocol provides us with a powerful performance benchmark: *actual human performance on the exact same situation* as encountered by the algorithm.

Furthermore, we partition this dataset into what we call “partial” and “full” trajectory datasets. In the partial trajectory dataset, we considered all (approximately) 10 meter long agent runs. For example, if agent 1’s full trajectory was 30 meters long, we would have 3 partial trajectories. Partial trajectory experiments provide focused examination of an algorithm’s ability to navigate through congestion in a safe and efficient manner. Full trajectory runs (the entire run



(a) (b) (c)

Figure 6: Full trajectory statistics. Total number of runs is 61; ; x, y axes in meters. All x-axes plot distance to nearest pedestrian; (a) plots means $\pm$ 1 standard deviation of algorithm and human; (b) appends plot (a) with d_{min} threshold, the closest distance two humans passed in the dataset. Inspection of region left of d_{min} shows numerous instances of DWA, MC_1e5, and SARL; (c) normalizes algorithm path length with human path length. E.g., values below $d_r = d_h$ mean the robot moved to goal more directly than human.

	$\mu(s)$	$\%_{s<0.3}$	$\%_{s<0.21}$	$\mu(d_r)$	$\%_{d_r/d_h>1.25}$	$\max(d_r/d_h)$	$\mu(t)$	$\mu(\max(\rho))$
Human	$.89 \pm .1m$	5%	5%	$25.6 \pm 6.2m$	NA	NA	NA	.22±.15
IGP_Full	$.55 \pm .17m$	15%	8%	$22.6 \pm 5.6m$	3%	1.4	$.2 \pm .3s$	.23± .12
IGP_Diag	$.57 \pm .2m$	30%	16%	$25.0 \pm 5.8m$	0%	1.2	$.11 \pm .07s$	.43± .4
IGP_Lin	$.55 \pm .2m$	26%	15%	$26.3 \pm 6.5m$	16%	1.4	$.12 \pm .1s$	.46 ± .9
DWA	$.4 \pm .15m$	50%	32%	$31.9 \pm 8.3m$	44%	2.2	$.1 \pm .03s$	.23 ± .1
MC_1e5	$.4 \pm .17m$	47%	31%	$32.7 \pm 8.6m$	57%	2.1	$4.7 \pm 3.2s$	.2 ± .3
SARL*	$.36 \pm .12m$	47%	25%	$8.9 \pm 2.7m$	0%	1.23	$1.4 \pm .2s$	.17 ± .1

*SARL was trained in five different environments; we report the best performing network.

Table 2: Full trajectory metrics. For each run, distance to nearest pedestrian is stored as s and $\mu(s)$ is the mean; $\%_{s<0.3}, \%_{s<0.21}$ are the percent of runs that $s < 0.3m, 0.21m$; $\mu(d_r)$ is the mean of the robot path length d_r over all runs; $\%_{d_r/d_h>1.25}$ is percent of times that d_r was 1.25 times the human path length d_h; $\mu(t)$ is the mean time of all replanning steps; and $\max_{runs}(\rho)$ is the maximum density observed. Density is computed in a 3m radius circle centered on robot position (in $people/m^2$).

of an agent) provide an additional level of information: they quantify how well the planner transitions between a variety of scenarios—strong interaction (head on avoidance, orthogonal crossing, following, being followed, etc), light interaction (a nearby person) and no interaction (straight line runs), all of which may occur in a single full trajectory run. We identified 293 partial trajectories and tested 181 (discarding 112 for calibration reasons). We tested 61 of the 150 full trajectories (discarding 89 trajectories for calibration reasons).

Since all algorithms replan from scratch at each step, long horizon planning (full trajectory) is straightforward: simply allow the goal at each time step to be the next step in the human's trajectory. We do not address the intricacies of combining long and short horizon planning here, and we are aware that providing this much human information can potentially bias the robot to over-perform. We guard against this potential confound with the partial trajectory studies (where only start and goal is provided). Also, performance in the full trajectory tests varies significantly across 6 different approaches.

Rationale for test algorithms We collected safety and path length data on humans, foIGP (IGP_Full), foIGP using the diagonals of Equation 5.1 (IGP_Diag), IGP with a linear prediction model for the agents (IGP_Lin), the "dynamic window approach" (DWA, (Fox, Burgard, and Thrun 1997), IGP using Monte Carlo optimization (MC_1e5, e.g., draw 10^5 random samples and choose the sample with largest Equation 6.1 value), and "socially aware reinforcement learning" (SARL, (Chen et al. 2019).

Each algorithm was chosen to explore a certain aspect of the performance space. We collected data on humans to serve as an upper bound on performance. We tested data on DWA for two reasons: First, this algorithm is widely deployed; in particular, it is the default navigation algorithm in ROS (see http://wiki.ros.org/base_local_planner). Thus it is a useful benchmark for many practitioners. Second, DWA is susceptible to known crowd navigation failure modes, such as freezing robot behavior (Trautman and Krause 2010). We chose MC_1e5 to test whether foIGP's optimization routine achieves better performance. We chose IGP_Lin to demonstrate that our inference approach can be used with *any* Gaussian mixture model (e.g., physics or machine learning based). We tested SARL because it outperforms all existing deep reinforcement learning approaches.

Finally, we trained SARL in 5 different environments. First, we trained in a 3m by 10m corridor (mimicking ETH conditions) with 15 and 5 people (0.5 and 0.16 people$/m^2$ densities), with mostly cross human cross traffic. The high density environment produced freezing robot behavior ($\%_{d_r/d_h>1.25} = 71\%$, $\max(d_r/d_h) = 23.8$), while the low density training produced a policy that was unsafe ($\%_{s<0.21} = 21\%$). We thus attempted training in the high density corridor, but with random start and goal positions of the people; this again resulted in freezing robot behavior ($\%_{d_r/d_h>1.25} = 18\%$, $\max(d_r/d_h) = 14.7$). We also trained in a 4m radius circular environment but with 10 people (density ≈ 0.2 people$/m^2$); in (Chen et al. 2019) a 4m radius circular environment with 5 people was the training environment. For reference, the average density in the ETH data was ≈ 0.2 people$/m^2$. The high density circular training also showed freezing robot behavior ($\%_{d_r/d_h>1.25} = 10\%$, $\max(d_r/d_h) = 4.32$). The training regimen of (Chen et al. 2019) thus produced the best policy. We tested this policy because it outperforms all existing deep reinforcement learning crowd navigation approaches, as well as "optimal reciprocal velocity obstacle" (ORCA) planning (van den Berg, Lin, and Manocha 2008) as shown in (Chen et al. 2019).

Partial and full trajectory safety and efficiency metrics We report the results of 181 partial trajectory runs in Figure 5, Table 1, and 61 full trajectory runs in Figure 6, Table 2. The $\mu(s)$ and $\mu(d_r)$ columns of the tables correspond to the ellipses in Figures 5, 6; the $\%_{s<.21}$ column is the percent of runs to the left of the d_{min} line in Figures 5, 6; the $\%_{d_r/d_h}$ column is the percent of runs above the $y = 1$ horizontal line in Figures 5(c), 6(c); $\max(d_r/d_h)$ is the largest y value in Figures 5(c), 6(c); $\mu(t)$ is replan time; and $\mu(\rho)$ is the mean of the maximum density of each run. Density is the number of people in a 3m radius circle around the robot ($people/m^2$).

The most important observation is revealed in Figures 5, 6 and in the $\%_{s<0.3}, \%_{s<.21}$ and $\%_{d_r/d_h>1.5}$ columns of Tables 1, 2: safety and path length of foIGP is competitive with humans. Additionally, the tables provide information about outlier values. For instance, the safety threshold $\%_{s<.21}$ for foIGP is nearly as good as human performance, but foIGP often chooses a shorter path than the human. foIGP requires $\approx 0.23s$ replan time (in python with a 2.6 GHz Intel Core i7 CPU and 16 GB RAM).

DWA, foIGP_Lin, MC_1e5, and SARL are not competitive with human performance. Notably, SARL has the shortest path lengths, but also has a very small safety margin (1 standard deviation almost reaches d_{min}, the human safety threshold determined empirically from the dataset), relatively high $\%_{s<.21}$ and among the highest $\%_{s<.3}$. Importantly, we trained SARL in five different environments and chose the highest performing network. None of these training environments improved upon the original training regimen. It remains unclear how to improve SARL's performance.

For DWA and MC_1e5 the values $\%_{d_r/d_h}$ and $\max(d_r/d_h)$ are very large. These algorithms exhibited freezing robot behavior, taking evasive paths to avoid congestion. For DWA this is unsurprising. For MC_1e5, this is likely because solutions that represent "cooperative collision avoidance" are unlikely to be found with vanilla sampling. Additionally, DWA and MC_1e5 had large values for $\%_{s<0.21}$. This is consistent with freezing behavior: in congestion they take evasive action that can force collisions.

8 Conclusion

Although this study makes progress on resolving the freezing robot problem, understanding valid forms of cooperative collision avoidance, and introduces a new method to recover locally optimal solutions of a complex joint distribution in real time, it must be emphasized that no real world experiments have taken place. This is a crucial asterisk on these results, since many additional factors are at play when robots are deployed in unscripted human environments. Further, the ETH dataset is static—that is, the agents are non-responsive—so whether or not a statistically valid CCA will port to real environments has yet to be decided. Additionally, crucial metrics like legibility and predictability can only be judged by human participants. To address this shortcoming, we are constructing a large scale, longitudinal study of robot navigation in unscripted human environments. We intend to test in multiple public spaces over a multi-year period.

Given the stated shortcomings of static testing, perhaps a good way to interpret simulation or canned dataset studies is as one of *invalidation:* we cannot validate that a crowd navigation algorithm will work in the real world, but we can show that an algorithm most likely will not work. For instance, DWA does not have the machinery for real world environments: in this study (and others), DWA exhibits freezing robot behavior and too high of a collision rate to risk deployment. SARL did not show freezing robot tendencies, although the collision rate in this study was too high to merit deployment (the SARL authors demonstrated the robot in a small crowd, but no performance metrics or operational parameters were reported). Given that SARL's performance was borderline, perhaps a different reward function would improve the safety to an acceptable tolerance.

While a real world study is the most pressing next step, theoretical issues remain. For instance, the Gaussian assumption is misaligned with human passing preferences; humans typically prefer passage on either side. However, our model of flexibility as covariance matrix assumes that left and right passing preference is identical. Further, our flexibility model is impoverished; better methods of understanding human "willingness to compromise" can surely be devised. Finally, while we tested on full trajectories, our algorithms were provided the global plan. Rigorously incorporating our current short horizon methods into a long horizon planner is nontrivial. Further, understanding how to plan over long distances given the native complexity of crowds is a substantial undertaking.

References

Adams, E. 2017. To survive the streets, robocars must learn to think like humans. *Wired*.

Alahi, A.; Goel, K.; and et al. 2016. Social-LSTM: Human trajectory prediction in crowded spaces. In *CVPR*.

Aoude, G., and et al. 2011. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion. *Autonomous Robots*.

Burgard, W.; Cremers, A.; and et al. 1998. The interactive museum tour-guide robot. In *AAAI*.

Canny, J., and Reif, J. 1987. New lower bound techniques for robot motion planning problems. In *Symposium on Foundations of Computer Science*.

Chen, Y.; Everett, M.; Liu, M.; and How, J. 2017. Socially aware motion planning with deep reinforcement learning. In *IROS*.

Chen, C.; Liu, Y.; Kreiss, S.; and Alahi, A. 2019. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *ICRA*.

Curtis, S.; Best, A.; and Manocha, D. 2016. Menge: A modular framework for simulating crowd movement. *Collective Dynamics*.

Doucet, A., and Johansen, A. 2008. A tutorial on particle filtering and smoothing: Fifteen years later. In *Oxford Handbook of Nonlinear Filtering*. Oxford Handbooks.

Du Toit, N., and Burdick, J. 2012. Robot motion planning in dynamic, uncertain environments. *IEEE Transactions on Robotics*.

Fan, T.; Cheng, X.; Pan, J.; Long, P.; Liu, W.; Yang, R.; and Manocha, D. 2019. Getting robots unfrozen and unlost in dense pedestrian crowds. *RAL*.

Fox, D.; Burgard, W.; and Thrun, S. 1997. The dynamic window approach to collision avoidance. *IEEE RAM*.

Fulgenzi, C., and et al. 2007. Dynamic obstacle avoidance in uncertain environment combining pvos and occupancy grid. In *ICRA*.

Fulgenzi, C.; Tay, C.; Spalanzani, A.; and Laugier, C. 2008. Probabilistic navigation in dynamic environment using rapidly-exploring random trees and Gaussian processes. In *IROS*.

Gloor, C. 2016. Pedsim: pedestrian crowd simulation.

Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; and Alahi, A. 2018. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *CVPR*.

Hall, E. 1966. *The Hidden Dimension*. Doubleday.

Helbing, D., and Molnar, P. 1995. Social force model for pedestrian dynamics. *Physical Review E*.

Ivanovich, B., and Pavone, M. 2019. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *ICCV*.

Ivanovich, B.; Schmerling, E.; Leung, K.; and Pavone, M. 2018. Generative modeling of multimodal multi-human behavior. In *IROS*.

Joseph, J.; Doshi-Velez, F.; and Roy, N. 2011. A bayesian nonparametric approach to modeling mobility patterns. *Autonomous Robots*.

Kretzschmar, H.; Spies, M.; Sprunk, C.; and Burgard, W. 2016. Socially compliant mobile robot navigation via inverse reinforcement learning. In *IJRR*.

Kruse, T.; Alami, R.; Pandey, A.; and Kirsch, A. 2013. Human-aware robot navigation: a survey. In *RAS*.

Kuderer, M.; Kretzschmar, H.; Sprunk, C.; and Burgard, W. 2012. Feature-based prediction of trajectories for socially compliant navigation. In *RSS*.

Lerner, A.; Chrysanthou, Y.; and Lischinski, D. 2007. Crowds by example. In *Computer graphics forum*.

Mead, R.; Atrash, A.; and Mataric, M. 2011. Proxemic feature recognition for interactive robots: automating social science metrics. In *ICSR*.

Pellegrini, S.; Ess, A.; Schindler, K.; and Van Gool, L. 2009. You'll never walk alone: modeling social behavior for multi-target tracking. In *International Conference on Computer Vision*.

Pettre, J.; Babel, M.; Hayet, J.; Salaris, P.; and Salvini, P. 2018. From freezing to jostling robots: Current challenges and new paradigms for safe robot navigation in dense crowds.

Pfeiffer, M.; Paolo, G.; Sommer, H.; Nieto, J.; Siegwart, R.; and Cadena, C. 2018. A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments. In *ICRA*.

Rasmussen, C. E., and Williams, C. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Rios-Martinez, J., and et al. 2011. Understanding human interaction for probabilistic autonomous navigation using risk-RRT. In *IROS*.

Sadigh, D.; Landolfi, N.; Sastry, S.; Seshia, S.; and Dragan, A. 2018. Planning for cars that coordinate with people: Leveraging effects on human actions for planning and active information gathering over human internal state. *Autonomous Robots*.

Snape, J.; van den Berg, J.; Guy, S.; and Manocha, D. 2011. The hybrid reciprocal velocity obstacle. *IEEE Transactions on Robotics*.

Sturlaugson, L., and Sheppard, J. 2014. Inference complexity in continuous time bayesian networks. In *UAI*.

Svenstrup, M.; Bak, T.; and Andersen, J. 2010. Trajectory planning for robots in dynamic human environments. In *IROS*.

Tai, L.; Zhang, J.; Lin, X.; and Burgard, W. 2018. Socially compliant navigation through raw depth inputs with generative adversarial imitation learning. *ICRA*.

Thompson, S., and et al. 2009. A probabilistic model of human motion and navigation intent for mobile robot path planning. In *IROS*.

Thrun, S.; Beetz, M.; and et al. 2000. Probabilistic algorithms and the interactive museum tour-guide robot minerva. *IJRR*.

Trautman, P., and Krause, A. 2010. Unfreezing the robot: Navigation in dense interacting crowds. In *IROS*.

Trautman, P.; Ma, J.; Krause, A.; and Murray, R. M. 2013. Robot navigation in dense crowds: the case for cooperation. In *ICRA*.

Trautman, P. 2013. *Robot Navigation in Dense Crowds: Statistical Models and Experimental Studies of Human Robot Cooperation*. Ph.D. Dissertation, California Institute of Technology.

Trautman, P. 2017. Sparse Interacting Gaussian Processes: Efficiency and Optimality Theorems of Autonomous Crowd Navigation. CDC.

Unhelkar, V., and et al. 2015. Human-robot co-navigation using anticipatory indicators of human walking motion. In *ICRA*.

van den Berg, J.; Lin, M.; and Manocha, D. 2008. Reciprocal velocity obstacles for real-time multi-agent navigation. In *ICRA*.

Vemula, A.; Muelling, K.; and Oh, J. 2018. Social attention: Modeling attention in human crowds. In *ICRA*.

Ziebart, B. D.; Maas, A.; and et al. 2008. Navigate like a cabbie: probabilistic reasoning from observed context-aware behavior. In *UbiComp*.